

Alona Otazgi

alona.otazgi@ehu.es In-reeh, endoso n-ik

Asisten 3h (1/5 kreditu)

Praktika (1/5 kreditu)

Asistents 8:00-10:00

12:00-13:00

Asistents 10:00-13:00

T.L. Floyd

Fundamentos de Sistemas Digitales

2^a ed. Prentice Hall, 2000

1. → Omenten kontzeptuak

2. → Boolean algebra

3. → TTL eta CMOS teknologia

4. → Sist digitalen deskriptzen moduak (Karnaugh mapak)

5. → Moduluz garatutako logikatu konbinatuak

948017430

Fernando Martinez

11 solasua

1. → Instrumentazioaren erabilera

2. → Ale logikatu: funtzio logikatu diseinua eta simulazioa

3. → Diseinu logikatuaren prozeduraren aplikazioak

5. → VHDL programazioa: konbinatu, konparatzaileak, multiplexadoreak, BCD/7 segmentu dekodagailuak, dekodagailu de multiplexadoreen artekoak.

6. → VHDL programazioa: konbinatu, USC-en erabilera

A taldea 20301

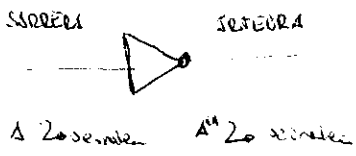
25 solasua ED lab

B taldea 20303 taldea bakoitza 3 asistents behar

Taldea bakoitza 4 solasua

C taldea 20301 Bikoitza eragozgarri berrak (10 taldeak)

Otsailak 25 ean hasten da A taldea



70. Carbonylale \rightarrow d'nerenken A-ien bebbenen neygen da

70 sek. erhaltet \rightarrow A' nervenlos etc. erhaltet A' von beiden nervenlos etc.
Menschen elementar behaltete 70 sek. erhaltet, bsp.

<u>Chemie</u>	<u>Symbol</u>	<u>Atomgewicht</u>	<u>Elektronenkonfiguration</u>
<u>Hydrieren</u>	10	0, 1, 2, 3, 4, 5, 6, 7, 8, 9	(128_{10})
<u>Bikar</u>	2	0, 1	(101101_2)
<u>Zink</u>	8	0, 1, 2, 3, 4, 5, 6, 7	(76_8)
<u>Hexadecanale</u>	16	0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15	$(126F_{16})$ Betreten, einbauen, beherrschen, H Zustand, blicken, ein

Homogeneous bilinear: $wxyt = w \cdot 0^3 + x \cdot 0^2 + y \cdot 0 + t \cdot 0^0$

3. 2. 2019

n. Džito kopura

Herbalkulturen - 07-1

B.ter. d.14 hennedersnaker a bitustikel, lounaker hesther d.14 aftrak. Aib: $\frac{100}{16} \times 100 = 28_{10}$
(skizette
hasiba)

B. Interne Kostenrechnung: interne Kostenrechnung. Ziel: Kosten der Leistungserstellung feststellen (Kostenrechnung).

Gray kedok esutan d senbaki batiok basteria, murengura, adakata bakura egan beherdela. Makin, suca
eta adakata sutraga, adan eta zara sutraga

1 — 0 0 0 1
2 — 0 0 1 1
3 — 0 0 1 0
4 — 0 1 1 0

BCD systemet afb-kode berekner idetten af bitene $\rightarrow 128_{10} = 0001\ 0010\ 1000$

BCD geht 3 systeme infra berechnen 3 geübten sätze

10 digits decimal or 26 letters = 36 ~~bits~~ letters, hence 2^6 better sigs (64)
(bits)

Wahrere 2^k bit unabhngig voneinander, nur mit konstante gefhrdung

Positives zu negatives exakt in die MSB abbilden da: 0 → positiv, 1 → negativ

ED II - 007

$0'5125 \cdot 2 = 0'625 \rightarrow 0$
 $0'625 \cdot 2 = 1'25 \rightarrow 1$
 $0'25 \cdot 2 = 0'5 \rightarrow 0$
 $0'5 \cdot 2 = 1 \rightarrow 1$

Erweitern, Multiplikation: $+$, \cdot , $-$, $=$

Erweitern, Logik: $+$, \cdot , \neg , \rightarrow , \leftrightarrow

\downarrow AND
 \downarrow OR
 \downarrow NOT

A	B	C	L
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	0
1	1	0	0
1	1	1	1

$$L = A \cdot B \cdot C$$

Aussagenlogik

- Ein logisches System ist ein System, das logische Aussagen behandelt.

- Die Aussagenlogik ist ein System, das logische Aussagen behandelt.

minterm

$$F = a \cdot b \cdot c + a \cdot b \cdot \bar{c} + a \cdot \bar{b} \cdot c + a \cdot \bar{b} \cdot \bar{c} + \bar{a} \cdot b \cdot c + \bar{a} \cdot b \cdot \bar{c} + \bar{a} \cdot \bar{b} \cdot c + \bar{a} \cdot \bar{b} \cdot \bar{c}$$

minterm

$$F = (a+b+c) \cdot (a+b+\bar{c}) \cdot (a+\bar{b}+c) \cdot (a+\bar{b}+\bar{c}) \cdot (\bar{a}+b+c) \cdot (\bar{a}+b+\bar{c}) \cdot (\bar{a}+\bar{b}+c) \cdot (\bar{a}+\bar{b}+\bar{c})$$

Erweitern, Logik, Aussagenlogik

- Logik ist ein System, das logische Aussagen behandelt.

- Minterm \rightarrow eine Aussage, die nur aus einer Aussage besteht \rightarrow 1 Aussage ist ein Minterm, 2 Aussagen sind ein Minterm.

- Minterm \rightarrow eine Aussage, die nur aus einer Aussage besteht \rightarrow 0 Aussage ist ein Minterm, 1 Aussage ist ein Minterm.

Aus






A	B	C	F
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	0
1	1	1	1

minterm

$$f = A \cdot B \cdot C + A \cdot B \cdot \bar{C} + A \cdot \bar{B} \cdot C + A \cdot \bar{B} \cdot \bar{C} + \bar{A} \cdot B \cdot C + \bar{A} \cdot B \cdot \bar{C} + \bar{A} \cdot \bar{B} \cdot C + \bar{A} \cdot \bar{B} \cdot \bar{C}$$

minterm

$$f = (A+B+C) \cdot (A+B+\bar{C}) \cdot (A+\bar{B}+C) \cdot (A+\bar{B}+\bar{C}) \cdot (\bar{A}+B+C) \cdot (\bar{A}+B+\bar{C}) \cdot (\bar{A}+\bar{B}+C) \cdot (\bar{A}+\bar{B}+\bar{C})$$

OR \rightarrow  \rightarrow 1 hat beide, ist eine 1 in der.
 AND \rightarrow  \rightarrow 0 hat beide, ist eine 0 in der.
 NOT \rightarrow  \rightarrow ist eine 0, ist eine 1 in der.
 NOR \rightarrow  \rightarrow 1 hat beide, ist eine 0 in der.
 NAND \rightarrow  \rightarrow 0 hat beide, ist eine 1 in der.

XOR \rightarrow OR exklusiv



XOR bedeutet beide, ist eine 0
 XOR bedeutet beide, ist eine 1

Ártek

$$1. A \cdot B + A(B+C) + B(B+C)$$

$$2. A \cdot \bar{B} + A \cdot (\bar{B}+C) + B(\bar{B}+C)$$

$$3. [A \cdot \bar{B} (C+B \cdot D) + \bar{A} \bar{B}] \cdot C$$

$$4. \bar{A} B \cdot C + A \cdot \bar{B} \bar{C} + \bar{A} \bar{B} \bar{C} + A \bar{B} C + A B \cdot C$$

$$5. \overline{A \cdot B + A \cdot C + A \cdot \bar{B} \cdot C}$$

$$6. f(A, B, C, D) = \sum (0, 1, 2, 3, 8, 15)$$

$$7. f(A, B, C) = \sum (2, 3, 4, 5)$$

$$8. \overline{A \cdot B + A \cdot C + \bar{A} \cdot \bar{B} \cdot C}$$

$$9. f(A, B, C) = \sum (3, 6, 7)$$

$\sum \rightarrow$ minimum

$\prod \rightarrow$ maximum

Egyszerűsítések

Három változós bázisú 0-1-ek tára (biterméni)

Lev változós bázisú 0-1-ek tára (biterméni)

Adat

A	B	C
0	0	0
0	0	1
0	1	0
0	1	1
1	0	0
1	0	1
1	1	0
1	1	1

$$1) A \cdot B + A \cdot B + A \cdot C + B \cdot B + B \cdot C$$

$$AB + AC + B + BC$$

$$B(A+1+C) + AC$$

$$\rightarrow \boxed{B + AC}$$

$$2) A \cdot \bar{B} + A \bar{B} \bar{C} + B \bar{B} \bar{C}$$

$$A \bar{B} (1 + \bar{C}) \rightarrow \boxed{A \bar{B}}$$

$$3) A \bar{B} [(C+B \cdot D) + 1] \cdot C \rightarrow \boxed{A \bar{B} \cdot C}$$

$$4) (A + \bar{A}) \bar{B} \bar{C} + A \bar{B} C + A \bar{B} \bar{C} + A B C$$

$$\bar{B} \bar{C} + B C (\bar{A} + \bar{A} + A) \rightarrow \boxed{\bar{B} \bar{C} + B C}$$

$$5) \overline{A \cdot B \cdot A \cdot C + A \cdot B \cdot C}$$

$$(\bar{A} + \bar{B}) \cdot (\bar{A} + \bar{C}) + \bar{A} \bar{B} C$$

$$\bar{A} + \bar{A} \bar{C} + \bar{B} \bar{A} + \bar{B} \bar{C} + \bar{A} \bar{B} C$$

$$\bar{A} (1 + \bar{C} + \bar{B} + \bar{B} C) + \bar{B} \bar{C} \rightarrow \boxed{\bar{A} + \bar{B} \bar{C}}$$

$$8) (\bar{A} \bar{B}) + (\bar{A} \bar{C}) + \bar{A} \bar{B} \cdot C$$

$$(\bar{A} + \bar{B}) \cdot (\bar{A} + \bar{C}) + \bar{A} \bar{B} C$$

$$\boxed{\bar{A} + \bar{B} \bar{C}}$$

$$\bar{A} + \bar{A} \bar{C} + \bar{B} \bar{A} + \bar{B} \bar{C} + \bar{A} \bar{B} C$$

$$\bar{A} (1 + \bar{C} + \bar{B} + \bar{B} C) + \bar{B} \bar{C}$$

$$9) f(A, B, C) = \sum (3, 6, 7)$$

A	B	C
0	0	0
0	0	1
0	1	0
0	1	1
1	0	0
1	0	1
1	1	0
1	1	1

$$\text{minimum } \bar{A} B C + A B \bar{C} + A B C$$

$$B C (\bar{A} + A) + A B \bar{C}$$

$$B (C + A \bar{C})$$

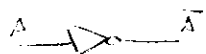
$$\boxed{B (C + A)}$$

$$\text{maximum } (A+B+C)(A+B+\bar{C})(A+\bar{B}+C)(A+\bar{B}+\bar{C})(\bar{A}+B+C)(\bar{A}+B+\bar{C})(\bar{A}+\bar{B}+C)(\bar{A}+\bar{B}+\bar{C})$$

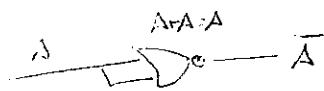
$$\text{maximum } (A+B+C)(A+B+\bar{C})(A+\bar{B}+C)(A+\bar{B}+\bar{C})(\bar{A}+B+C)(\bar{A}+B+\bar{C})(\bar{A}+\bar{B}+C)(\bar{A}+\bar{B}+\bar{C})$$

NOR dleak mchiltha

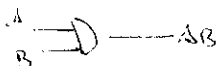
NOT



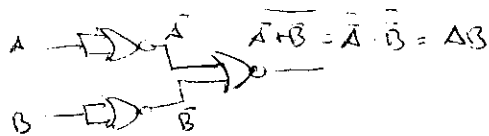
< >



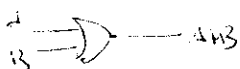
AND



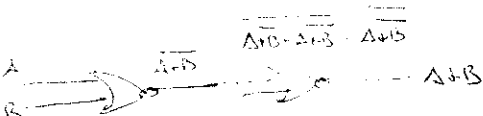
< >



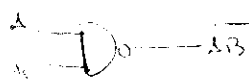
OR



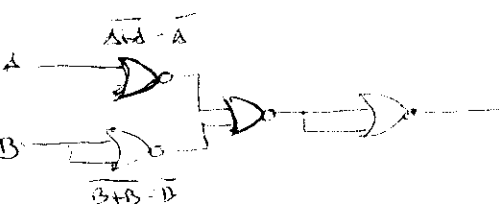
< >



NAND



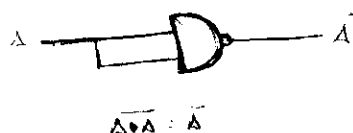
< >



THE LOGIC OF "NOR" GATE "NAND" ATAK ERABJULST

NOT

NAND



NOT

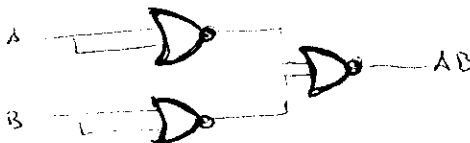
NOR



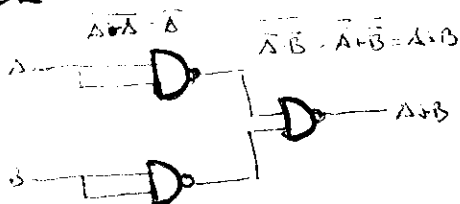
AND



AND



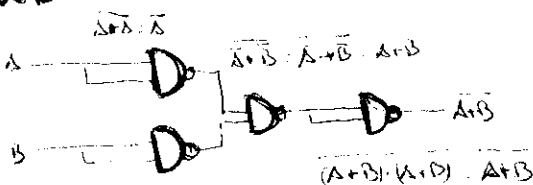
OR



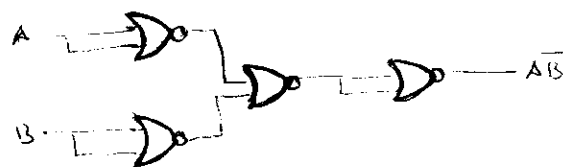
OR



NOR



NAND



$$f = (A+B+C)(A+\bar{B}+C)(\bar{A}+\bar{B}+C)(\bar{A}+B+C)$$

000 010 110 101

maximieren $\bar{A} \rightarrow 1$
 $A \rightarrow 0$

AB \ C	0	1
00	0	
01	0	
10	0	
11		0

! Ablesen hat fallen vergessen, b1 bekommt energie diktieren.

Ab.B

$$f(A,B,C) = \bar{A} + A\bar{B} + ABC$$

000 100 110
 001 101
 010 101
 011

AB \ C	0	1
00	1	1
01	1	1
10	1	
11	1	1

$$f(A,B,C,D) = \bar{B}\bar{C} + A\bar{B} + ABC + A\bar{B}C\bar{D} + \bar{A}B\bar{C}D + A\bar{B}CD$$

0000 1000 1101 1010 0001 1011
 0001 1001 1100
 1000 1010
 1001 1011

AB \ CD	00	01	11	10
00	1	1		
01				
11	1	1		
10	1	1	1	1

Ablesen hat fallen über beide d1re kolumne

2'-kolumne

Teileren benenne $\rightarrow 2^{abg} \cdot k$

Teileren benenne = kolumne kolumne

AB \ C	0	1
00	1	1
01	1	
10		1
11	1	1

AB \ CD	00	01	11	10
00	1	1		
01	1	1	1	1
11				
10			1	1

AB \ CD	00	01	11	10
00	1			1
01	1	1		1
11	1	1		1
10	1			1

! Simplifikation herange greifen

6 aldehyd reagent, bromdehyd kerdike dthyo eke leu teub mullkatu, teub he-rek kerdike aldehyd an eke k... b... g... b... d... (00, 01, 11, 10). Anreketan berale, taldeak beure beketen eginge dthyo eke m... ane.

teketen esogabete, funktiek

A	B	C	D	f
0	0	0	0	0 (0)
0	0	0	1	0 (1)
0	0	1	0	0 (2)
0	0	1	1	0 (3)
0	1	0	0	0 (4)
0	1	0	1	0 (5)
0	1	1	0	0 (6)
0	1	1	1	0 (7)
1	0	0	0	1 (8)
1	0	0	1	1 (9)
1	0	1	0	1 (10)
1	0	1	1	1 (11)
1	1	0	0	1 (12)
1	1	0	1	1 (13)
1	1	1	0	1 (14)
1	1	1	1	1 (15)

Kee beketen b... d... b... 0 di 1 ber, keu haer X p... d... beu egi k... beu k... en.

Adibden, tektu b... sareu berale BCD kerdike 7, 8, 9 kerdike seherakan interan 1 eneng duen funktiek an disetua.

! No eke anreketan kasuak et k... agertu, d... beu beu BCD kerdike.

Se reakt aldehyd bet aldehyd interan eke m... m... beu beu → eke eke
Se reakt aldehyd bet aldehyd interan alatu beu beu → eke dinamika

estetik

A	B	f
0	0	0
0	1	0
1	0	1
1	1	1

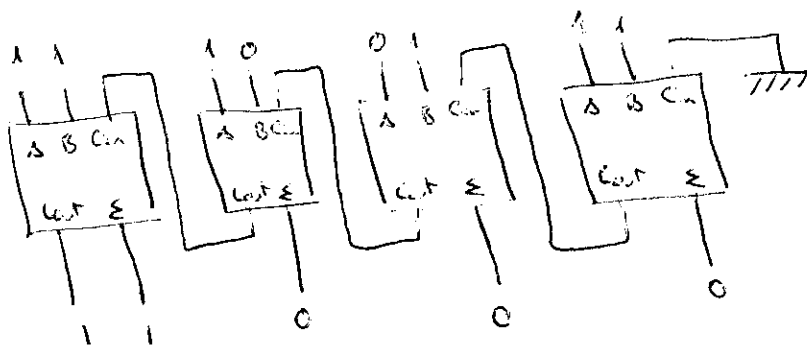
dinamika

A	B	f
0	0	0
0	1	1
1	0	1
1	1	0

EV-54-017

Ans/ $A = 1101$
 $B = 1011$

$$\begin{array}{r} 1101 \\ 1011 \\ \hline 11000 \end{array}$$



Fast-achiever

Bewakings SORRALS: Betreffende betrekkingen inzake bewaking en het dienen, welke samen met 2 betrekkingen worden de bewaking.

Buruknya HEDSPEN secara langsung intermediet dengan. Secara tidak langsung ada banyak lainnya.

2 digitale 2 kerbaxi baxheke

$$1. \text{etagen} \rightarrow c_i = \underbrace{g_i}_{1} + \underbrace{p_0 c_i}_{0}$$

2. $\text{eth}_{\text{per}} \rightarrow C_2 = g. + p.c. = g_2 + p.(g_2) = \underline{g. + p.g_2}$

! Etapek en dit lunde he bidden bekansik,
beukle skavinske en soveleke ere
jachten dte.

24.5/

$$1000_{10} = 999_{10} + 999_{10}$$

$$3 \text{CCl}_4 \rightarrow \text{HCCl}_3 + 2 \text{CCl}_2$$

- ↳ **Konstitutiv** ein **statistisches**, **hypothesen**
integriertes **Wissenschafts** **disziplin**
sozialwissenschaftliche **disziplin**

$\underbrace{\quad\quad\quad}_{3 \text{ integral}} \quad \underbrace{\quad\quad\quad}_{2 \text{ integral}} \quad \underbrace{\quad\quad\quad}_{1 \text{ integral}}$

1. Keşif (discovery) - Keşif (discovery) - Keşif (discovery)

Bedeutung: Konkrete Inhalte

$$c) (1+5) - (1+2)$$

from bita — 0 → 1
 1 → 1

$-S: 0101 \rightarrow 0010 \rightarrow 1011$

10110-1101-1002-2

$$\frac{1110}{1100} \rightarrow 26 \text{ oxygen} \rightarrow 1100 \xrightarrow{126 \text{ oxygen}} 1011 \rightarrow 7$$

n. Agosto 1892.

$$[2^{n-1}, 2^n - 1]$$

Walter - 4 - [8.7]

! Nicht korrekter kann es nicht sein
manche, wenn man es so
genau die meisten kulturellen
in der Lage. Ist das genug
behalten werden.

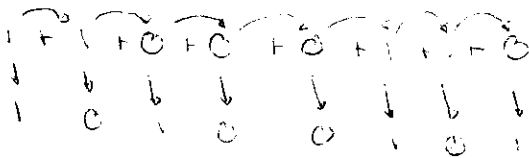
74148: Serien- oder Parallel-Addierer
 4-Bit-Block, erweiterte Variante

Ein 4-Bit-Input Modulo-2-Addierer
 liefert die 4-Bit-Parallel-Addition
 der beiden 4-Bit-Operanden.

Ein 4-Bit-Output Modulo-2-Addierer
 liefert die 4-Bit-Parallel-Addition
 der beiden 4-Bit-Operanden.

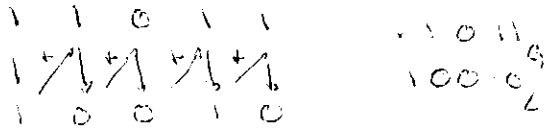
4-Bit-System-Addierer

Bit-Gray / Gray-Code



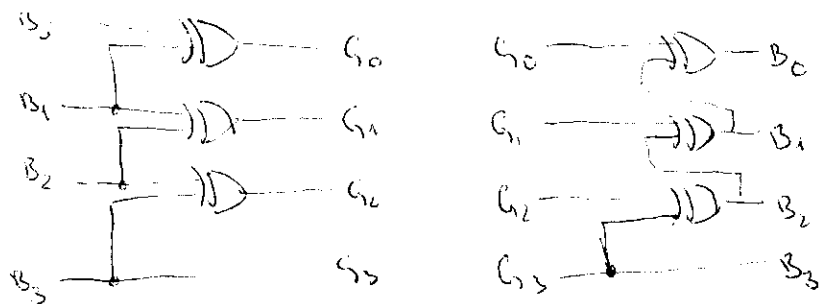
11000102

10100104



Bit-Gray (4-Bit-Gray)

$$\begin{aligned} G_0 &= B_0 \oplus B_1 & B_3 &= G_3 \\ G_1 &= B_1 \oplus B_2 & B_2 &= G_2 \oplus G_3 \\ G_2 &= B_2 \oplus B_3 & B_1 &= G_1 \oplus B_2 \\ G_3 &= B_3 & B_0 &= G_0 \oplus B_1 \end{aligned}$$



Verbit die 4-Bit-Gray-Code in den 4-Bit-Binary-Code? 7.1.1.1

$$P_8 = P_1 \oplus P_2 \oplus D_3 \oplus P_4 \oplus D_5 \oplus D_6 \oplus D_7$$

Pariteitsbit berechnen:

$$P_8 = 0 \rightarrow \text{bit kann beliebig gewählt werden}$$

$$C_8 = 1 \text{ bewirkt, bit als ausgehend zu setzen}$$

$$C_8 = 0 \text{ bewirkt, bit als einlaufend zu setzen}$$

$$C_8 = P_1 \oplus P_2 \oplus D_3 \oplus P_4 \oplus D_5 \oplus D_6 \oplus D_7 \oplus P_8$$